IN THE CLAIM

1. (Previously Amended Once, Currently Amended) A method for retiring a first instruction[s] processed through various processing stages, comprising the steps of:

processing a second instruction different from the first instruction;

processing ~~an~~ the first instruction ~~capable of early retirement~~ until the first instruction meets the criteria for early retirement;

indicating that the first instruction has met the early-retirement criteria;

processing the first instruction to a desirable stage, and at which, based on an ~~the~~ indication that the first instruction has met the early-retirement criteria, terminating the first instruction ~~is terminated~~ out of order of a program running the first instruction; and

updating a state of a system processing the first instruction to reflect that the first instruction has been terminated;

wherein the early-retirement criteria is met when processing the second instruction causes ~~the instruction is processed to a point~~ that continued processing of the first instruction does not change ~~the~~ an architectural state of the system processing the first instruction, and, at the time of termination, the first instruction has completed its function without completing its full pipeline.

2. (Currently Canceled)

3. (Original) The method of claim 1 wherein the step of indicating comprises the step of generating a signal associated with the instruction.

4. (Currently Amended) The method of claim 3 further comprises the steps of:

sending the signal to an early-retirement unit; and

the early-retirement unit arranging for the first instruction to be terminated.

5. (Original) The method of claim 1 wherein the various processing stages include one or more of the following stages: fetching, issuing, sorting, executing, queuing, and retiring.

6. (Currently Amended) The method of claim 1 wherein [the] an instruction capable of early retirement includes an identification tag for identifying whether the instruction is capable of early retirement.

7. (Currently Canceled)

8. (Previously Canceled)

9. (Currently Canceled)

10. (Previously Amended Once, Currently Amended) A computer-readable medium embodying instructions that cause a computer to perform a method for retiring a first instruction[s] processed through various processing stages, the method comprising the steps of:

processing a second instruction different from the first instruction;

processing ~~an~~ the first instruction ~~capable of early retirement~~ until the first instruction meets the criteria for early retirement;

indicating that the first instruction has met the early-retirement criteria;

processing the first instruction to a desirable stage, and at which, based on an ~~the~~ indication that the first instruction has met the early-retirement criteria, terminating the first instruction ~~is terminated~~ out of order of a program running the first instruction; and

updating a state of a system processing the first instruction to reflect that the first instruction has been terminated;

wherein the early-retirement criteria is met when processing the second instruction causes ~~the instruction is processed to a point~~ that continued ~~execution~~ processing of the first instruction does not change ~~the~~ an architectural state of the system processing the first instruction, and, at the time of termination, the first instruction has completed its function without completing its full pipeline.

11. (Currently Canceled)

12. (Original) The computer-readable medium of claim 10 wherein the step of indicating comprises the step of generating a signal associated with the instruction.

13. (Currently Amended) The computer-readable medium of claim 12 wherein the method further comprises the steps of:

sending the signal to an early-retirement unit; and

the early-retirement unit arranging for the first instruction to be terminated.

14. (Original) The computer-readable medium of claim 10 wherein the various processing stages include one or more of the following stages: fetching, issuing, sorting, executing, queuing, and retiring.

15. (Currently Amended) The computer-readable medium of claim 10 wherein ~~the~~ an instruction capable of early retirement includes an identification tag for identifying whether the instruction is capable of early retirement.

16. (Currently Canceled)

17. (Previously Canceled)

18. (Currently Canceled)

19. (Previously Amended Once, Currently Amended) A system for retiring a first instruction[s] processed through various processing stages, comprising at least one means for:

~~first processing means for~~

processing a second instruction different from the first instruction;

processing ~~an~~ the first instruction ~~capable of early retirement~~ until the first instruction meets the criteria for early retirement;

~~second processing means for~~ processing the first instruction to a desirable stage, and at which, based on an indication that the first instruction has met the early-retirement criteria, terminating the first instruction ~~is terminated~~ out of order of a program running the first instruction; and

~~updating means for~~ updating a state of the system to reflect that the first instruction has been terminated;

wherein the criteria for early retirement is met when processing the second instruction causes ~~the instruction is processed to a point~~ that

continued processing of the first instruction does not change ~~the~~ an architectural state of the system, and, at the time of termination, the first instruction has completed its function without completing its full pipeline.

20. (Currently Canceled)

21. (Currently Added) The method of claim 1 wherein processing the second instruction comprises executing a branch condition of a branch instruction.

22. (Currently Added) The method of claim 21 wherein terminating the first instruction includes discarding execution of a branch of the branch instruction.

23. (Currently Added) The method of claim 1 wherein processing the second instruction comprises loading a register with a value that the register already contains.

24. (Currently Added) A method for retiring an instruction processed through various processing stages, comprising the steps of:

processing the instruction until the instruction meets the criteria for early retirement;

indicating that the instruction has met the early-retirement criteria;

processing the instruction to a desirable stage, and at which, based on an indication that the instruction has met the early-retirement criteria, terminating the instruction out of order of a program running the instruction; and

updating a state of a system processing the instruction to reflect that the instruction has been terminated;

wherein the early-retirement criteria is met when the instruction is identified as

performing a logical OR of a register with the same register, writing a value 0 from a register into the same register, or writing a value to a register hardwired to a predetermined value.

25. (Currently Added) The method of claim 24 further comprises one or a combination of the following steps to determine whether the early-retirement criteria is met:

evaluating an op-code of the instruction; and

evaluating a register target of the instruction.

26. (Currently Added) A method for retiring early a NO-OP instruction that is scheduled to be processed through various processing stages, the method comprising the steps of:

evaluating an op-code of an instruction to determine whether the instruction is the NO-OP instruction, and, if so, using a tag to indicate that the NO-OP instruction has met the early-retirement criteria;

processing the NO-OP instruction to a desirable stage, and at which, based on the indication that the NO-OP instruction has met the early-retirement criteria, terminating the NO-OP instruction out of order of a program running the NO-OP instruction; thereby skipping processing the NO-OP instruction in at least one of the processing stages; and

updating a state of a system processing the NO-OP instruction to reflect that the NO-OP instruction has been terminated.